

**МУНИЦИПАЛЬНОЕ КАЗЕННОЕ УЧРЕЖДЕНИЕ  
«УПРАВЛЕНИЕ ОБРАЗОВАНИЯ АДМИНИСТРАЦИИ ГОРОДА КАНСКА»  
МУНИЦИПАЛЬНОЕ БЮДЖЕТНОЕ УЧРЕЖДЕНИЕ  
ДОПОЛНИТЕЛЬНОГО ОБРАЗОВАНИЯ  
«ЦЕНТР ДЕТСКОГО ТЕХНИЧЕСКОГО ТВОРЧЕСТВА»**

Принята на заседании методического совета  
МБУ ДО ЦДТТ «20» марта 2015 г.

«Утверждаю»

Директор МБУ ДО ЦДТТ

\_\_\_\_\_ С.А. Руленко  
(подпись, печать)

**Общеобразовательная программа дополнительного образования  
«Программирование на языке Pascal»**

Программа рассчитана на учащихся 13 - 17 лет

Срок реализации программы: 1 год

Составил: Усачев Сергей Валерьевич  
педагог дополнительного образования ЦТТ

2015 г.

## Пояснительная записка

По уровню программа рассчитана на среднее (полное) общее образование, имеет научно – техническую направленность. Срок реализации программы - 1 год.

Программа ориентирована на обучающихся 13 - 17 лет, желающих заниматься программированием в будущем или же познакомиться с компьютерным программированием с целью подготовки к изучению этого предмета в высшем или средне-специальном учебном заведении. Обучение по данной программе не требует наличия у подростка компьютерной техники вне компьютерного класса, в котором проводятся занятия.

Программа способствует профессиональной ориентации подростков и их подготовке к получению специальности программиста, даёт возможность оценить свои перспективы в этой области.

**Цель программы:** освоение обучающимися основ компьютерного программирования на алгоритмических языках с использованием процедурно-ориентированной среды программирования Turbo Pascal.

### Задачи программы:

- Научить подростков основам алгоритмических языков программирования;
- Обучить воспитанников составлению алгоритмов;
- Создание условий для развития логического мышления обучающихся;
- Содействовать развитию навыков самоорганизации воспитанников, их уверенности в себе.

Программа «Программирование на процедурных языках» является модифицированной программой «Программирование», разработанного С.П.Митрофановым и С.Г.Проскуряковой, преподавателями Сургутской высшей гимназии – лаборатории Салахова в 2002 году.

Занятия проводятся два раза в неделю по два часа. Чтобы обучающиеся чувствовали себя более комфортно, группы формируются по возрасту – 7-9 и 10-11 классы. Количество участников в группе не превышает 10 человек.

Занятия делятся на лекционные (лекционно-практические), на которых тема изучается всей группой, и индивидуальные (индивидуально-практические), на которых и осваивается основная часть тем.

Целью обучения является закладывание основ для изучения алгоритмических (процедурных) языков программирования. И хотя программа предполагает не освоение какого-то определённого языка программирования, все практические задачи рассматриваются на примере языка Turbo Pascal 7.0. Данный язык программирования изначально был задуман как обучающий, он отражает фундаментальные идеи алгоритмов в легко воспринимаемой форме, позволяет четко реализовать идеи структурного программирования и структурной организации данных.

## Учебно-тематический план

Тема	Всего часов	Теория	Практика
Основные понятия программирования	12	4	8
Графические возможности языка Turbo Pascal	14	6	8
Прямые и разветвлённые алгоритмы	12	4	8
Циклические алгоритмы	14	6	8
Введение в целочисленную арифметику	12	4	8
Создание многомодульных программ	14	6	8
Рекурсивные алгоритмы	14	6	8
Одномерные массивы	14	6	8
Двумерные массивы	12	4	8
Алгоритмы сортировки массива	14	6	8
Строковый тип данных	12	4	8
<b>Итого:</b>	<b>144</b>	<b>56</b>	<b>88</b>

# Содержание программы

## 1. Основные понятия программирования (12 часов)

История появления языка Pascal и причины его популярности. Опции меню среды TP7. Структура программы, описание и преобразование типов, действия над типами, команды ввода и вывода информации.

*Основные понятия.* Тип переменной (integer, real, char, string, boolean), оператор begin..end, оператор :=, команды read, readln, write, writeln, позиционный вывод целых и вещественных чисел, стандартные процедуры и функции языка (div, mod, round, trunc, sqrt, sqr, sin, cos, tg, ln, lg, randomize, random, abs)

*Практическая работа* Составление программ на нахождение значений арифметических выражений, операции над числами при помощи процедур и функций.

## 2. Графические возможности языка Turbo Pascal (14 часов)

Познакомить учащихся с историей появления языка Pascal и причинами его популярности. Научить пользоваться опциями меню среды TP7. Познакомить учащихся со структурой программы, описанием и преобразованием типов, действиям над типами, командами ввода и вывода информации.

*Основные понятия.* Тип переменной (integer, real, char, string, boolean), оператор begin..end, оператор :=, команды read, readln, write, writeln, позиционный вывод целых и вещественных чисел, стандартные процедуры и функции языка (div, mod, round, trunc, sqrt, sqr, sin, cos, tg, ln, lg, randomize, random, abs)

*Практическая работа* Составление программ, создающих рисунки в среде Turbo Pascal. Программирование движения графических объектов.

## 3. Прямые и разветвлённые алгоритмы (12 часов)

Научить учащихся составлять алгоритмы, в которых используется выбор альтернативы. Показать, что существуют задачи, в которых необходимо использовать несколько условных операторов, вложенные условные операторы.

*Основные понятия.* Команды if..then; if..then..else, case, case..else. Полный, неполный, простой, составной условный оператор, логические связки (or, and, not), логические выражения. Вложенный условный оператор.

*Практическая работа* Составление программ, включающих в себя несколько условных операторов, вложенные условные операторы.

## 4. Циклические алгоритмы (14 часов)

Рассказ учащимся о необходимости использования циклических алгоритмов. Объяснение, как записываются, и как работают отдельные циклические команды. Показ отличия между циклами for, while, repeat. Демонстрация, как можно записать один и тот же фрагмент программы с использованием различных циклов. Научить решать задачи с использованием вложенных циклов. Объяснить, что неправильная запись команды может привести к заикливлению.

*Основные понятия.* Циклы for..do, while..do, repeat..until, параметр цикла, тело цикла, заикливление, условие выполнения (окончания) цикла, вложенные циклы.

*Практическая работа* Составление программ с использованием циклов for..do, while..do, repeat..until. Составление задач с использованием вложенных циклов.

## 5. Введение в целочисленную арифметику (12 часов)

Показ на примерах свойства натуральных чисел. Научить решать задачи.

*Основные понятия.* Простое, совершенное, автоморфное, дружественное, симметричное числа; числа близнецы, палиндромы, числа ряда Фибоначчи.

**Метод проверки:** турнир программистов. Учащимся предлагается придумать свои собственные программы по только что пройденному разделу. Оценивается стабильность, проработанность интерфейса, быстрота, простота для конечного пользователя, краткость и умелое оформление исходного кода программы. *Низкий уровень знаний* соответствует выполнению слегка видоизменённой задачи из рассмотренных на занятиях примерах. Программа нестабильна, интерфейс пользователя не проработан или отсутствует, исходный код не оформлен должным

образом. *Средний уровень знаний* соответствует выполнению придуманной самим учеником задачи по данному разделу. Программа стабильна, за исключением некоторых недочётов. Интерфейс пользователя присутствует. Исходный код оформлен должным образом. *Высокий уровень знаний* соответствует выполнению задачи, придуманной самим учеником, использующей знания и всех изученных им разделов. Программа полностью стабильна, интерфейс пользователя прост и понятен. Исходный код краток, правильно оформлен и снабжён комментариями. Решённые задания разбираются вместе с учителем индивидуально или в небольших группах. По итогам проверки выявляется победитель.

## **6. Создание многомодульных программ (14 часов)**

Объяснить, что такое метод пошаговой детализации и необходимость его применения в ряде задач. Рассказать учащимся, что такое процедуры и функции. Показать, как они записываются и чем они отличаются. Объяснить, в каких случаях удобно использовать процедуры, а в каких функции. Научить записывать рекурсивные алгоритмы.

*Основные понятия.* Метод пошаговой детализации, процедура, функция, формальные и фактические параметры, механизм передачи параметров по ссылке и по значению, глобальные и локальные переменные.

*Практическая работа* Составление программ с использованием собственных функций и процедур.

## **7. Рекурсивные алгоритмы (14 часов)**

Объяснить, что такое метод пошаговой детализации и необходимость его применения в ряде задач. Рассказать учащимся, что такое процедуры и функции. Показать, как они записываются и чем они отличаются. Объяснить, в каких случаях удобно использовать процедуры, а в каких функции. Научить записывать рекурсивные алгоритмы.

*Основные понятия.* Рекурсивный алгоритм, прямая и косвенная рекурсия, организация выделения стека.

*Практическая работа* Составление программ с использованием рекурсивных алгоритмов.

## **8. Одномерные массивы (14 часов)**

Познакомить учащихся со структурным типом данных - массивами. Показать преимущество использования массивов при решении ряда задач. Научить решать задачи с использованием алгоритмов обработки массивов с применением циклов.

*Основные понятия.* Тип массив, одномерный массив, элемент массива, индекс элемента.

*Практическая работа* Составление программ с использованием алгоритмов обработки массивов с применением циклов.

## **9. Двумерные массивы (12 часов)**

Ввести понятие двумерного массива. Дать отличия от одномерных массивов. Научить решать задачи с использованием двумерных массивов.

*Основные понятия.* Двумерный массив, строка и столбец массива.

*Практическая работа* Составление программ с использованием двумерных массивов.

## **10. Алгоритмы сортировки массива (14 часов)**

Познакомить учащихся с правилами сортировки массивов и основными методами сортировки.

*Основные понятия.* Сортировка массива. Методы сортировок массивов.

*Практическая работа* Составление программ с использованием сортировки вставкой, сортировки выбором, сортировки методом простого обмена, рекурсивной сортировки и сортировки методом слияний.

## **11. Строковый тип данных (12 часов)**

Познакомить учащихся с типом String и функциями, выполняемыми над этим типом. Научить решать задачи предложенные учителем.

*Основные понятия.* Тип String. Функции pos, insert, val, str, copy, delete, length.

*Практическая работа* Составление программ с использованием строкового типа данных.

**Подведение итогов года.** Возможны различные формы – викторины, конкурсы и т.д.

По завершению изучения тематических разделов программы проводятся зачетные занятия. Обучающиеся выполняют самостоятельные работы, в которых отвечают на контрольные вопросы и решают задачи в среде программирования Turbo Pascal. *Низкий уровень знаний* соответствует нескольким решённым заданиям с меткой **(Н)** или же одному решённому заданию с меткой **(С)**. *Средний уровень знаний* соответствует нескольким решённым заданиям с пометкой **(С)** или же одному решённому заданию с пометкой **(В)**. *Высокий уровень знаний* соответствует решению нескольких заданий с меткой **(В)**, нескольких заданий с меткой **(С)** и нескольких заданий с меткой **(Н)**. Решённые задания разбираются вместе с педагогом индивидуально или в небольших группах. Задания предоставляются в избыточном количестве для того, чтобы предоставить обучающимся право выбора вопроса, задания.

## **Результативность программы**

Результатом обучения является умение обучающихся составлять алгоритмы, писать небольшие программы на языке Turbo Pascal 7.0, проверять их работоспособность.

Подведение итогов по каждому пройденному разделу осуществляется посредством наблюдения за выполнением практических работ. А вот при подведении итогов года лучше использовать нестандартные формы, включающие в себя элемент игры: конкурс компьютерных программ, турнир программистов, викторина.

### **Прогнозируемые результаты обучения:**

#### ***Знания:***

- Общий подход к решению задач по программированию.
- Виды алгоритмов.
- Основы Турбо-Паскаля.
- Графические возможности Турбо-Паскаля.
- Понятие «хороший стиль программирования».
- Основы устройства и принципа организации компьютеров.

#### ***Умения:***

- Создавать программы на языке Турбо-Паскаль.
- Создавать программы, рисующие несложные картинки.
- Создавать читаемые программы (т.е. использовать «хороший стиль программирования»).
- Решать задачи с нечётко поставленным условием.
- Организовывать интерфейс с пользователем.
- Работать с компьютерной литературой.

#### ***Личностные качества:***

- Навыки самостоятельной работы.
- Дисциплинированность, общая организованность.
- Уверенность в работе с компьютером.
- Профессиональная подготовка к получению профессии программиста.
- Самостоятельность в выполнении полученных заданий.

## **Методическое обеспечение программы**

Основой программы являются теоретические и практические занятия. Для их проведения требуется разделить всю программу обучения на отдельные небольшие темы в соответствии с указаниями в разделе «Содержание программы». Теоретический материал по каждой теме содержится в отдельной презентации PowerPoint, которая формируется по следующему **принципу**:

- Теоретический материал изложен языком, понятным обучающимся.
- Теоретический материал снабжён примерами.

Для материала, касающегося изучения операторов, предлагается следующая **схема организации текста** с объяснениями:

- Общий вид оператора.

- Принцип работы оператора.
- Блок-схема оператора.
- Пример программы с использованием оператора.
- Пояснения к программе.
- Особенности оператора, возможные ошибки при использовании оператора.

Кроме теоретического материала, в письменном (печатном) виде необходимо иметь и все решаемые в процессе обучения задачи по программированию. Рекомендуется каждое задание выносить на отдельный лист бумаги (карточку), выдавая следующее задание лишь после правильного выполнения предыдущего. По каждой теме следует иметь задания как реализующие какие-то особенности изучаемого оператора или типа данных, так и задания, сходные по смыслу друг с другом.

#### ***Принцип формирования набора задач, выдаваемых обучающимся.***

При изучении каждой темы обучающийся обязательно должен получить и выполнить задания, реализующие основные особенности и случаи использования изучаемого явления. Например, при изучении темы «Одномерные массивы» воспитанник должен решить задачи, включающие нахождение максимального (или минимального) элемента, вычисление суммы (разности, произведения и т.д.) элементов, обмен элементов и так далее. Таким образом, для успешного освоения темы «Одномерные массивы» каждый обучающийся должен выполнить хотя бы одно задание на поиск максимального или минимального элемента, и хотя бы одну задачу, вычисляющую сумму, разность или произведение элементов и т.д. Подростки, у которых темпы освоения программы выше среднего, могут выполнить одно упражнение, включающее все перечисленные элементы.

Наличие нескольких сходных по смыслу задач позволит, во-первых, обеспечить индивидуальными задачами каждого обучающегося (снимается проблема списывания решений друг у друга) и, во-вторых, позволит педагогу в случае возникновения у обучающегося проблем с решением задачи, оказать ему существенную помощь (вплоть до решения задачи вместе с ним), после чего выдать подростку аналогичное задание. Такой подход позволит преподавателю проверить понимание каждым обучающимся методики решения задачи рассматриваемого типа.

С целью получения корректной информации о знаниях обучающихся не рекомендуется проводить заключительное занятие в конце года в виде контрольной работы. Лучшим решением будет использование нестандартных форм подведения итогов года, включающих в себя элементы игры. Выбор той или иной формы осуществляется педагогом, исходя из особенностей конкретной группы подростков.

Варианты проведения итоговых занятий, опробованных на практике:

#### *Конкурс компьютерных программ*

Важно, чтобы участие в конкурсе было добровольным, то есть свои программы представляли бы лишь те, кто сам выбрал для себя именно эту форму зачёта. С целью повышения образовательного эффекта рекомендуется дать возможность воспитаннику самостоятельно придумать задание к программе и лично эту программу разработать. Педагог в этом случае должен вмешиваться в работу обучающегося лишь при возникновении серьёзных осложнений в разработке проекта.

Работа над собственным (неучебным) проектом позволяет подростку лучше разобраться в пройденном материале, почувствовать себя программистом-профессионалом.

#### *Викторина*

Участники разбиваются на две команды, соревнующиеся друг с другом. В викторину входят несколько заданий:

- составление и решение кроссворда по информатике;
- решение задачи по известному условию и восстановление условия по предоставленному решению;
- печатание текста (на скорость).

### **Методика составления задач, решаемых учащимися**

Каждая задача должна быть такой, чтобы подросток мог решить её за одно занятие. Стандартные задания для учащихся должны, в основном, описывать возможности изучаемого оператора. Задачи повышенной сложности для обучающихся должны, описывая возможности оператора, давать ребенку представление о стандартных приёмах разработки программ.

Очень важно, чтобы задачи включали в себя элементы, изученные ранее. Такой подход позволит педагогу контролировать понимание изученного материала, а обучающемуся – вспомнить и лучше осознать пройденные темы, понять, что каждая изучаемая тема чрезвычайно важна для дальнейшего обучения.

### **Перечень необходимого оборудования, инструментов и материалов для реализации дополнительной образовательной программы.**

#### **Техническое и программное обеспечение**

Для реализации данной программы требуются IBM-совместимые компьютеры с процессором типа Intel 80286 и выше. Желательно соответствие между числом учащихся и числом компьютеров как 1:1. Для демонстрации лекционного материала требуется проектор.

На компьютерах должна быть установлена операционная система MS-DOS или Windows технологии '95 (Windows'95, Windows'98, Windows ME и т.д.) и язык программирования Turbo Pascal 7.0. На презентационном компьютере также должен быть установлен офисный пакет MicroSoft Office версии 97 и выше (или же аналогичный ему бесплатный программный продукт – Open Office версии 2.0 и выше).

#### **Список литературы**

1. Фаронов В.В. «Программирование на персональных ЭВМ в среде Турбо-Паскаль» – М.: Изд-во МГТУ, 580 стр. 1991 г.
2. Фигурнов В.Э. «IBM PC для пользователя. 5-е издание» – М.: Финансы и статистика, 480 стр. 1994 г.
3. Эллиот Б. Коффман «Turbo Pascal Web Update, 5th Edition», издательство Вильямс, 896 стр, 2005 г
4. Дональд Э. Кнут «Искусство программирования», издательство Вильямс, 720 стр, 2006 г
5. Немнюгин С.А. «Turbo Pascal. Программирование на языке высокого уровня», издательство Питер, 544 стр., 2005 г.
6. Коджаспирова Г.М., Коджаспиров А.Ю. «Словарь по педагогике (междисциплинарный)», издательство «МарТ», 447 стр, 2005 г
7. **www.pascal.ru** - Полный курс обучения языку программирования Turbo Pascal.